

提出日：2024 年 4 月 19 日

# 先進情報プロジェクト実習 I

テーマ IT・1 第 1 回レポート

学籍番号：C0A22113

氏名：成田 彩華

## ダウンロード

今回ダウンロードするものは Workstation 17 Player for Windows と ubuntu 22.04 LTS の iso ファイルの 2 つである。まずは Workstation 17 Player for Windows をダウンロードしていく。

Workstation 17 Player for Windows を以下の vmware の公式サイトからダウンロードする <https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html> 今回私は MacBook を使用するため vmware はサポート外である。そのため、UTM を使用する。

次に ubuntu 22.04 LTS をダウンロードしていく。以下の ubuntu の公式サイトから 4 つのミラーサイトのどれかを 1 つ選び、ダウンロードする。私は富山大学からダウンロードした。 <https://www.ubuntulinux.jp/download/ja-remix>

ARM64 である apple シリコンの MacBook を使用しているため、上記の ubuntu はインストールできない。そのため今回は ubuntu サーバーをインストールしたのち、ubuntu サーバーにデスクトップをインストールする形で対応する。

ubuntu 22.04 LTS のハッシュ値を確認し、授業サイトに書いてあるハッシュ値と一致するかどうかで、正しくダウンロードできているか確かめる。授業サイトからわかる通り、今回使用されているハッシュ関数は SHA-256 である。よってオプションで SHA-256 を指定してハッシュ値を求める必要がある。ダウンロードした iso ファイルのハッシュ値を確認するコマンドは以下の通りである。

```
shasum -a 256 <ダウンロードしたisoファイルのパス>
```

計算したハッシュ値と授業サイトに書いてあるハッシュ値が一致していれば、ダウンロード中にファイルが破損することなくダウンロードできていることとなる。

## VMware のインストール

VMware のインストールを行なっていく。

基本的にウィザード通りに進めて行く。互換性セットアップの画面では「Windows Hypervisor Platform (WHP)の自動インストール」にチェックを入れ「次へ」を押す。使用許諾書の画面では「使用許諾契約書に同意します」にチェックを入れ「次へ」を押す。カスタムセットアップの画面では「拡張キーボードドライバ」にチェックを入れて「次へ」を押す。ユーザーエクスペリエンスの設定の画面では「起動時に製品の更新を確認する」のチェックを外し「次へ」を押す。

Windows を再起動すると VMware のインストールが完了する

## Ubuntu のインストール

Ubuntu のインストールを進めていく。

まずは、VMware を起動する。初回起動時は非営利目的で、VMware Workstation 17 Player を無償で使用する(U)を選択して続行する。

「新しい仮想マシン ウィザード」の「新規仮想マシン作成ウィザードへようこそ」では「後で OS をインストール(S)」を選択して「次へ(N)」をクリックする。

「新しい仮想マシン ウィザード」の「ゲスト OS の選択」では、ゲスト OS に「Linux(L)」を選択し、バージョン(V)に「Ubuntu 64 ビット」を選択して「次へ(N)」をクリックする。

「新しい仮想マシン ウィザード」の「仮想マシンの名前」では、「仮想マシン名(V)」と「場所(L)」は任意のものでよい。ただし、覚えておくこと。面倒であれば、入力されている情報のままにしておく。

「新しい仮想マシン ウィザード」の「ディスク容量の指定」では、「ディスク最大サイズ(GB)(S)」をできるだけ大きい値にしておく。20GB では容量が厳しく、30GB 以上が無難と思われる。

「仮想ディスクを単一ファイルとして格納(O)」を選択しておく。パフォーマンスの著しい低下がみられる可能性があるため、使用するディスク容量が同じであれば単一ファイルを選んでおいたほうが無難である。

「新しい仮想マシン ウィザード」の「仮想マシンを作成する準備完了」では、「ハードウェアをカスタマイズ(C)」をクリックし、「この仮想マシンのメモリ(M)」の値を 4096MB より大きい値にしておくほうが快適である。大きく設定しておくことで、スワップがはいらないようにする。

「仮想マシン設定」の「ISO イメージ ファイルを使用する(M)」では、ダウンロードした ubuntu-ja-22.04-desktop-amd64.iso のファイルを指定する。

仮想マシンを再生して「GNU GRUB version 2.06」の画面が出たら、「Try or Install Ubuntu」を選択する。この画面では、Ctrl+G で仮想マシンにキーボードとマウスの制御が移る。Ctrl+Alt で Windows にマウスの制御が戻る。

Ubuntu の「インストール」という画面が出たら、「Ubuntu をインストール」を選択する。「Ubuntu を試す」ではないので注意する。

「インストール」の「キーボードレイアウト」では、各自のキーボードレイアウトを選択する。

「インストール」の「アップデートと他のソフトウェア」では、「通常のインストール」を選択しておく。「その他のオプション」は任意でよい。

「インストール」の「インストールの種類」では、「ディスクを削除して Ubuntu をインストール」を選択する。「これにより、すべての OS 上にあるプログラム、ドキュメント、写真、音楽、その他のファイルはすべて削除されます。」と恐ろしい警告が書かれているが、

これは仮想マシンであるため Windows は削除されない。

「インストール」の「あなたの情報を入力してください」で、以下の情報を聞かれるが、全角文字入力のはちに問題が発生する可能性があるため、半角で入力する。

あなたの名前 : ayaka

コンピュータの名前 : vm-ayaka

ユーザー名の入力 : ayaka

パスワードの入力 : ayaka787

パスワードの確認 : ayaka787

インストール後の、輪がぐるぐる回っている「... then press Enter」のような画面では Enter キーを押さないと次に行かない。

Ubuntu のセットアップが完了したらログインする。「オンラインアカウント」の設定は「スキップ(S)」でよい。

「Livepatch」の設定は「次へ(N)」にする。

「Ubuntu の改善を支援する」の設定は「いいえ、送信しません」を選択して「次へ(N)」にする。

「Ubuntu へようこそ」の設定は「次へ(N)」にする。

「準備完了」になったら「完了(D)」をクリックする。

「ソフトウェアの更新」が現れたらインストールしておく。

これで Ubuntu のインストールが完了である

## Apache のインストール

Web サーバとして Apache2 を使えるようにするため、Apache のインストールをしていく。

まず端末 (Terminal) を出す。画面左下に□が9個並んでいるところをクリックする。そこに「端末」があるのでクリックする。これでコマンドを入力できるようになる。Ctrl + Alt + T でも Terminal は起動する。

まず Ubuntu を最新の状態にする

```
$ sudo apt update
```

```
$ sudo apt upgrade -y
```

ソフトウェアの更新から Ubuntu 22.04 のリリース後に、ソフトウェアがアップデートされました。今すぐインストールしますか？でも最新の状態になる。

次に Apache のインストールをしていく

```
$ sudo apt install apache2
```

以下のように聞かれたら続行する。

この操作後に追加で 7,721 kB のディスク容量が消費されます。

続行しますか? [Y/n] Y

なお、今回は、port80 だけが使えば良い。port80 は Web サーバがクライアントからの接続を待ち受けて HTTP でデータを返信する標準のポートとして用いられているため、Web ブラウザでからサーバーにアクセスできるようにするためには port 80 が使えばよい。TLS/SSL は証明書を格好するにはグローバル IP 出ないと発行してもらえないので今回は HTTPS ではなく HTTP を使用する。

## Web サーバに Web ブラウザからアクセスする

Web サーバに Web ブラウザからアクセスするため、Ubuntu の IP アドレスを調べる。Ubuntu の IP アドレスは以下のコマンドでわかる

```
$ hostname -I  
192.168.xxx.xxx
```

xxx の部分はユーザごとにランダム割り当てである。私の場合は 192.168.64.7 であった。

このとき、Ubuntu ではなく Windows のブラウザから、Ubuntu の Web サーバにアクセスできる。

先ほど調べた IP アドレスを Windows のブラウザに打ち込み、アクセスする。「Apache2 Ubuntu Default Page」が表示されるのが確認できるはずである。

サブネットが異なる、ネットワークが異なる場合、ルータを使用し、ネットワークをつなげる。このパソコン自体の IP アドレスは 192.168.40.34 であり、ネットワークが異なる。私の環境では vmenet0 の IP アドレスが 192.168.64.1 となっており、これがルータがわりになり VM との通信を実現させていると考える。ローカル側も VM 側も ping 192.168.64.1 でパケットが確認できている。

## Apache2 の仕組み

/var/www/html が WWW のドキュメントルートなので、この下にディレクトリを作成しファイルをおいてみる。Web ブラウザにサーバの IP アドレスのみを入力しすると /var/www/html が参照される。

apache の設定ファイルは /etc/apache2/apache2.conf、環境設定ファイルは /etc/apache2/envvars である。設定ファイル自体は root で読み書きしている。

```
$ ls -al /etc/apache2

-rw-r--r-- 1 root root 4096 4月12 15:47 .
-rw-r--r-- 1 root root 4096 4月12 15:47 ..
drwxr-xr-x 2 root root 4096 4月12 17:16 conf-available
drwxr-xr-x 2 root root 4096 4月12 16:50 conf-enabled
-rw-r--r-- 1 root root 1782 12月5 03:58 envvars
-rw-r--r-- 1 root root 31063 12月5 03:58 magic
drwxr-xr-x 2 root root 12288 4月12 15:47 mods-available
drwxr-xr-x 2 root root 4096 4月12 16:50 mods-enabled
-rw-r--r-- 1 root root 320 12月5 03:58 ports.conf
drwxr-xr-x 2 root root 4096 4月12 15:47 sites-available
drwxr-xr-x 2 root root 4096 4月12 15:47 sites-enabled
```

/etc/apache2/envvars の中身を確認してみる。

```
# These need to be set in /etc/apache2/envvars

User ${APACHE_RUN_USER}

Group ${APACHE_RUN_GROUP}


$ cat envvars | grep APACHE_RUN

export APACHE_RUN_USER=www-data

export APACHE_RUN_GROUP=www-data

export APACHE_RUN_DIR=/var/run/apache2${SUFFIX}
```

つまり、www-data がユーザ名でグループ名である。web ブラウザからファイルにアクセスする際は www-data がファイルにアクセスする。

## Web コンテンツのアクセス権限

まず、ドキュメントルートに test ディレクトリを作成する。

```
$ mkdir /var/www/html/test
```

/var/www/html/index.html を /var/www/test にコピーをする

```
$ cp /var/www/index.html /var/www/html/test
```

/var/www/html/test 内のアクセス権限を考える。

```
ayaka@vm-ubuntu:/var/www/html/test$ ls -al
合計 20
drwxr-xr-x 2 ayaka ayaka 4096 4月 12 04:53 .
drwxr-xr-x 3 root  root  4096 4月 12 04:31 ..
-rw-r--r-- 1 ayaka ayaka 10918 4月 12 04:53 index.html
```

これだと Web ブラウザ上から test ディレクトリ、index.html にはアクセスできるが危険。test ディレクトリで他ユーザが読み込みと実行権限持っているため、www-data 以外でも読み込みと実行ができてしまう。

```
ayaka@vm-ubuntu:/var/www/html/test$ ls -al
合計 20
drwx----- 2 ayaka ayaka 4096 4月 12 04:53 .
drwxr-xr-x 3 root  root  4096 4月 12 04:31 ..
-rw----- 1 ayaka ayaka 10918 4月 12 04:53 index.html
```

これだと Forbidden You don't have permission to access this resource. と表示される。web ブラウザからアクセスされるときは www-data がアクセスするが、所有者、グループに www-data がいなく、パーミッションも test と index.html 共にその他ユーザには何も許可されていないため、アクセスできない。

```
ayaka@vm-ubuntu:/var/www/html/test$ sudo ls -al
合計 20
drwx----- 2 www-data www-data 4096 4月 12 04:53 .
drwxr-xr-x 3 root  root  4096 4月 12 04:31 ..
-rw----- 1 www-data www-data 10918 4月 12 04:53 index.html
```

これであれば test ディレクトリ、index.html 共に所有者である www-data のみアクセスできるパーミッション設定であるため、アクセスでき、安全である。

## ファイルの編集方法とフォームの原理

Web に公開するファイルを編集する。

```
$ gedit ex13_1_ans.html
```

これでテキストエディタを使用してファイルの編集ができる。

このファイルの中身は HTML で書かれている。項目はリストのタグである<li>が用いられており、type="disc"で黒丸アイコンを指定し、並べている。

例

```
<li type="disc">サンプルテキスト</li>
```

各<li>タグの項目の中に<input>タグがあり、type を指定してあげることで、テキスト入力、パスワード入力、隠し入力、複数行のテキスト入力、ラジオボタン入力、チェックボックス入力、ファイル送信を設置している。

最後に<input>タグで送信ボタンとリセットボタンを設置し、この全体を<form>タグで囲うことで、フォーム形式となり、ページが完成している。この入力をサーバーに送った後、action で指定した cgi ファイルなどを実行する形となる。

## CGI の設定

授業ページにある test.cgi を動かせるようにする。cgi ファイルを動かすには Perl が必要となる。よってどこに Perl があるのか調べる。

```
$ which perl
```

```
/usr/bin/perl
```

と出力される。よって CGI ファイルの 1 行目を以下のものに修正する。

```
#!/usr/bin/perl
```

これで CGI ファイルの変更が完了した。

次に Apache2 で CGI を使えるようにする。

/etc/apache2/conf-available/cgi-enabled.conf ファイルを作成する。このファイルの中身は以下の通りである。

```
<Directory "/var/www/html/test">

    Options +ExecCGI

    AddHandler cgi-script .cgi .pl

</Directory>
```

次のコマンドで変更を反映させる。

```
$ sudo systemctl reload apache2
```



```
$ sudo a2enconf cgi-enabled
$ sudo a2enmod cgi
$ sudo systemctl restart apache2
```

もし「500 Internal Server Error」となったら、CGI ファイルを置いたディレクトリのパーミッション、CGI ファイルのパーミッションを 755 に変更する。755 は所有者に読み書き、グループとその他に読み取り専用の権限に設定することを表す。それでもエラーが出てしまったら、改行コードが対応していない場合がある。1 行目の Perl のパスを以下のように変更する。

```
#!/usr/bin/perl -
```

さらに、CGI ファイル内の改行コードの修正をする。修正するためのコマンドは以下の通りである。

```
$ sudo cat test.cgi | tr -d “\r” > test2.cgi
```

これで改行コードが削除された。削除された CGI ファイルが test2.cgi に保存されている。

Web ブラウザに 192.168.64.7/test/test2.cgi と打ち込むと test2.cgi の実行結果が表示される。

## 掲示板 CGI プログラムの設置

授業ページにある bbs.cgi を動かせるようにする。まず、bbs.cgi には BOM が付いている。この BOM を消す必要がある。消すにはまず、改行コードを UNIX のものにする。

```
$ sudo cat bbs.cgi | tr -d “\r” > bbs.cgi
```

次に、gedit で「名前をつけて保存」し、文字コードを UTF-8 にする。

```
$ sudo gedit bbs.cgi
```

さらに nkf で BOM を消す。nkf が入っていないのでまずは nkf をインストールする。

```
$ sudo apt install nkf
```

```
$nkf -overwrite -oc=UTF-8 bbs.cgi
```

bbs.cgi のパーミッションは上記の test.cgi 同様、755 で設定する。所有者に読み書き権限、グループとその他に読み取り専用権限を与える。

ログファイルの設置を行う。log.txt を test ディレクトリに設置する。所有者を www-data にし、パーミッションは所有者に読み書き権限を与えれば良い。グループとその他のユーザには権限を与えなくて良い。

## Python で CGI を動かしてみる

python でも CGI を動かす。test ディレクトリ内に py\_test.cgi という名前で CGI ファイルを作成する。ファイルの中身は以下の通りである。

```
#!/usr/bin/python3

print("Content-Type: text/plain\n")

print('Hello, World!')
```

Web ブラウザに 192.168.64.7/test/py\_test.cgi と入力すると CGI ファイルの実行結果が表示される。今回は Hello, World! と表示される。